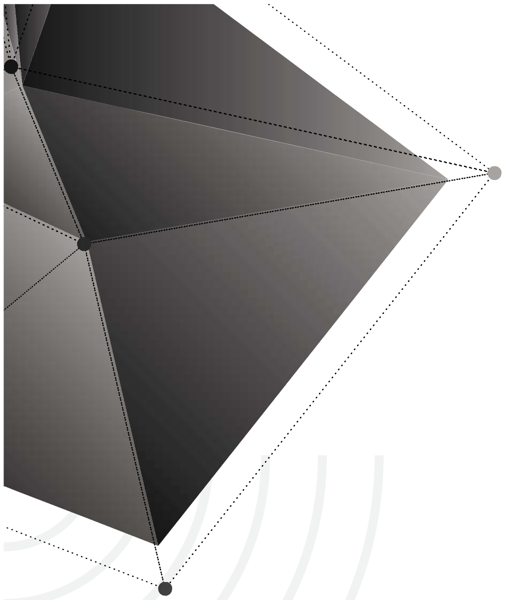
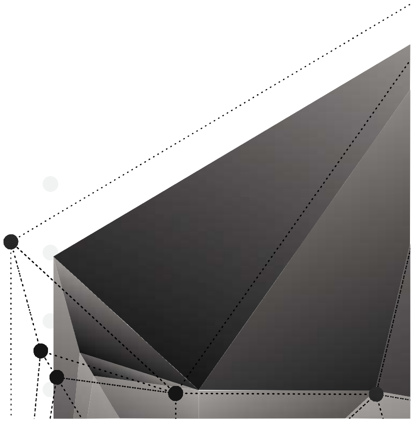


FOURTH SEMESTER

'COMMUNICATION & COMPUTER NETWORKING'



DIGITAL COMMUNICATION SYSTEMS

Course Code	474001
Course Title	Digital Communication Systems
No. of Credits	5 (TH:4,T:0,P:2)

COURSE OUTCOME: After completion of this course the students are able to:

1. Understand the fundamental principles and concepts of digital communication systems.
2. Analyze the block diagrams and subsystems of various modulation techniques, such as PCM, DPCM, DM, and ADM.
3. Evaluate the performance of different modulation techniques in terms of quantization noise and low bit rate coding for speech and video signals.
4. Comprehend the concept of baseband transmission, matched filtering, and the effects of additive Gaussian noise on system performance.
5. Apply the Nyquist criterion for zero Inter Symbol Interference (ISI) and utilize sinusoidal roll-off filtering, correlative coding, and equalizers to mitigate ISI.
6. Compare and contrast different digital modulation schemes, including OOK, BPSK, coherent and non-coherent FSK, QPSK, DPSK, QAM, MSK, and multicarrier modulation, in terms of bandwidth and bit rate.

COURSE CONTENTS

Unit - 1 :

- Block diagram & brief description of a digital communication system.
- Pulse code modulation (PCM): Basic block diagram.
- Differential pulse code modulation (DPCM): Basic block diagram and its working.
- Delta Modulation (DM): Basic block diagram & its working.
- Adaptive Delta Modulation: Basic block diagram & its working.

Unit - 2 :

- Quantization noise of Delta & Adaptive Delta Modulation.
- Baseband transmission, Inter symbol interference (ISI), Nyquist criterion for zero ISI, Equalizers, Digital subscriber lines (DSL).

Unit - 3 :

- Coherent and non-coherent FSK, comparison of bandwidth and bit rate of digital modulation schemes.

Unit - 4 :

- Need and method of multiplexing: TDMA, FDMA, CDMA (block diagram and their comparison)

Unit - 5 :

- Communication Channel characteristics: bit rate, baud rate, bandwidth.
- Introduction to Information and Coding Theories: Basics of Information Theory, Shannon entropy, linear block codes–definitions, properties, types.

PRACTICAL OUTCOMES: Upon completion of the course, student will be able to:

1. Generate PCM Signal and recover original signal from PCM Signal.
2. Testing of DPCM modulation and demodulation.
3. Testing of DM modulation and demodulation.
4. Understand the TDM frame structure.
5. Use CDMA technique to transmit signal.

List Of Practicals:

1. Test the performance of PCM Generation and Demodulation of analog signal.
2. Test the performance of Differential Pulse Code Modulation and Demodulation Technique.
3. Test the performance of Delta Modulation
4. Test the performance of Adaptive Delta modulation.
5. Study of CDMA technique using analog signal as an input signal.
6. Study of Time Division Multiplexing.
7. Study of QAM transmitter and receiver.

Text & Reference Books:

1. Communication Systems Haykin, S 4th Ed., John Wiley & Sons
2. Modern Digital and Analog Communication SystemsLathi, B.P. and Ding, Z Intl. 4th Ed., Oxford University Press.
3. Digital Communications Proakis, J.G. and Saheli, M 5th Ed., McGraw-Hill
4. Digital Communication:Fundamentals and Applica Sklar, B., and Ray, P.K 2nd Ed., Dorling Kindersleytions
5. T. Cover and J. Thomas 2/e, Wiley.
6. Principles of Digital Communication, R. G. Gallager Cambridge Univ. Press
7. A Foundation in Digital Communication A. Lapidoth Cambridge Univ. Press
8. Error Control Coding S. Lin and D. Costello 2/e, Prentice Hall.

*****

ADVANCE NETWORKING

Course Code	474002
Course Title	Advance Networking
No. of Credits	5 (TH:4,T:0,P:2)

COURSE OUTCOMES: Upon completion of the course, students will be able to:

1. Gain understanding of internet working basics, including Ethernet cabling, IP addressing, and troubleshooting techniques.
2. Master the art of IP sub netting and Variable Length Subnet Masks (VLSM) to efficiently manage and allocate IP addresses.
3. Acquire knowledge about routing basics, including static and dynamic routing, to establish effective network communication.
4. Develop expertise in switching basics, including configuring layer 2 switches and implementing secure and scalable VLANs.
5. Learn how to manage network traffic using Access Control Lists (ACLs) for improved security and performance.

COURSE CONTENTS

1. Internet working Basics

Internet working Basics, Broadcast domain, Internet Protocols: - TCP/IP Model, IP Addressing, IP Terminology, IP Addressing Scheme, Private IP Addresses, TCP/IP Troubleshooting utilities, Troubleshooting IP Addressing.

2. IP Subnetting & Variable Length Subnet Masks (VLSM)

Subnetting Basics, How to Create Subnets, Subnet Masks, Basics of Classless Inter- Domain Routing (CIDR), Variable Length Subnet Masks (VLSMs).

3. Routing

IP Routing, Routing Basics, Static Routing, Default Routing, Dynamic Routing, Routing Protocol (Basics).

4. Switching

Concept of Layer 2 Switching, Configuring the L2 Switches, L2 Switch Startup, Setting the Passwords, Setting the Hostname, Setting IP Information, Configuring Interface Descriptions, Erasing the Switch Configuration.

5. VLAN

Virtual LAN Basics, Broadcast Control, Security, Flexibility and Scalability, Types, Frame Tagging, LAN Identification Methods, Inter- Switch Link (ISL) Protocol. VLAN Trunking Protocol (VTP), Routing between VLANs, Configuring VLANs.

6. Managing Traffic with Access Control Lists

Managing Traffic with Access Lists, Introduction to Access Lists, Standard Access Lists, Extended Access Lists.

7. WAN Protocols

Introduction of WAN, Cabling the WAN, Frame Relay, ISDN, DSL/ADSL.

8. Introduction To Wireless LAN

ISM band, IEEE 802.11 standard for WLAN, 802.11a/b/g wireless standards Adhoc, infrastructure mode of WLAN, Access Point in Repeater Mode, Security in WLAN.

PRACTICAL OUTCOMES: Upon completion of the course, student will be able to:

1. Understand the configuration and management of routers and switches for different LAN segments.
2. Gain proficiency in IP routing by creating & managing static routes.
3. Acquire the skills to configure VLANs and perform inter-VLAN routing.
4. Learn to manage network traffic using standard IP access lists.
5. Develop the ability to configure Network Address Translation (NAT) for efficient IP address usage.
6. Gain practical experience in setting up and managing wireless LAN.

LIST OF PRACTICALS

1. To study Router & its interface. (Console port, AUI, Serial, Auxiliary, Ethernet, Fast Ethernet, BRI)
2. To study, Switch & its interface. (Console port, Ethernet, Fast Ethernet)
3. To setup up a router, logging into a router, basic commands, saving NVRAM configuration.
4. To configure a router for different LAN segments.
5. To configure IP routing by creating static routes. (static routing)
6. Backing Up and Restoring the IOS, Configuration File using TFTP server
7. To Setup up a Switch first time, logging into a switch, basic commands
8. To configure VLANs and Inter-VLAN Routing.
9. To manage traffic using standard IP Access list.
10. Configuring Static NAT on router
11. Configuring Dynamic NAT on router
12. Wireless LAN Setup using ADHOC mode.
13. Wireless LAN Setup using Infrastructure mode.

Text & Reference Books:

1. RFCs and Standards Documents (www.ietf.org and other standard body websites)
2. Communication Networking – An Analytical Approach, Anurag-Manjunath-Joy
3. TCP/IP Illustrated (Vol.1,2), Stevens
4. Data Networks, Bertsekas-Gallager
5. An Engineering Approach to Computer Networking, S. Keshav

Websites for Reference:

1. http://swayam.gov.in
2. http://spoken-tutorial.orgs

*****

ELECTRONIC MEASUREMENT AND INSTRUMENTATION

Course Code	474003
Course Title	Electronic Measurement & Instrumentation
No. of Credits	4 (TH:4,T:0,P:0)

COURSE OUTCOMES: Upon completion of the course, students will be able to:

1. Select proper measuring instrument with suitable accuracy.
2. Maintain different instruments.
3. How to calculate value of resistance, capacitance, inductance using different bridges.
4. Explain different types of potentiometers.
5. Explain the working of measuring instruments.
6. Work in a CRO, and measure different quantities by CRO.
7. Explain the working of Transducers.

COURSE CONTENTS

Unit – I : Basics of Measurements and Bridges

- Accuracy & precision, Resolution
- Types of Errors
- DC Bridges – Wheatstone Bridge
- AC Bridges - Maxwell's Bridge, Hay's Bridge

Unit - II : Potentiometer

- Basic DC Potentiometer & its applications
- AC Potentiometer & its applications

Unit– III : Measuring Instruments (Construction & Working)

- Permanent Magnet Moving Coil Instruments (PMMC)
- Moving Iron type Instruments (MI)
- Electro Dynamo Type Instruments
- Single Phase Energy Meter

Unit– IV : Electronic Instruments (Working Principle)

- Analog Voltmeter and Digital Voltmeter
- Electronic Multimeters
- Q – Meter

Unit– V : Oscilloscopes and DSO

- Basic block diagram of Cathode ray oscilloscope and working of its various sub-systems.
- Measurement of frequency, time delay, phase angle and modulation index.
- Basic block diagram of digital storage oscilloscope (DSO) and working of its various sub-systems.

Unit - VI : Transducers

- Classification, Characteristics, Construction, Working Principles and Application of following Transducers.
- RTD, Thermocouple, Thermistor, LVDT, Strain Gauge, Load Cell.

Suggested Learning Resources:

1. Electrical & Electronic Measurement & Instruments A.K. Sawhney, Dhanpat Rai & Sons, India
2. Electronic Instrument and Measurement Technique, W.D. Cooper, Prentice Hall International, India.
3. Electronic Measurement & Instrumentation, J.G. Joshi, Khanna Publishing House, Delhi
4. Measurement systems application and design, E.O. Doebelin and D. N.Manik, The Mcgraw-Hill
5. Electronic Measurements and Instrumentation, Oliver and Cage, The Mcgraw-Hill
6. Basic Electrical Measurement, M.B. Stout, Prentice hall of India, India
7. Electronic Instrumentation, H. S. Kalsi, The Mcgraw-Hill

*****

INTRODUCTION TO OPERATING SYSTEMS

Course Code:	434003
Course Title	Introduction to Operating System
No. of Credits	5 (TH:4,T:0,P:2)

COURSE OUTCOMES: After completion of this course, student will be able to:

1. Understand the basic concepts and architecture of UNIX/LINUX operating systems.
2. Gain knowledge of process management, including process concepts, operations, and inter-process communica-tion.
3. Learn about different process scheduling algorithms and their implementations.
4. Develop an understanding of memory management techniques such as allocation, swapping, paging, and segmentation.
5. Explore file management concepts, including file access methods, directory structure, and file system implementation.
6. Acquire knowledge of OS security principles, including authentication, access control, and system logs.

COURSE CONTENTS

UNIT - 1:

1.1 Overview of Operating System

1.2 Basic concepts

1.3 UNIX/LINUX Architecture

1.4 Kernel

1.5 Services and systems calls

1.6 System programs.

UNIT - 2 :

2.1 Overview of Process management Concept

2.2. Process scheduling Methods

2.3 Multi- threaded programming

2.4 Memory management Techniques

2.5 Virtual memory

UNIT - 3 :

3.1 File management

- 3.1.1 Concept of a file
- 3.1.2 Access methods

3.2 Directory structure

3.3 Overview of File system structure and implementation

3.4 Different types of file systems

UNIT - 4 :

4.1 I/O system

4.2 Mass storage structure

4.2.1 Overview

4.2.2 Disk structure

4.2.3 Disk attachment

4.3 Basic Idea of Disk scheduling algorithms

4.4 Swap space management

4.5 Raid.

UNIT - 5:

5.1 OS Security

5.2 Authentication

5.3 Access Control

5.4 Access Rights

5.5 System Logs

Practical Outcomes: Upon completion of the course, student will be able to:

1. Simulate CPU scheduling algorithms (FCFS, SJF, Round Robin, Priority) using a C program.
2. Implement the producer-consumer problem simulation using semaphores in C.
3. Simulate the Dining-philosophers problem to understand synchronization and resource allocation.
4. Implement memory allocation techniques (MVT, MFT) in a simulation.
5. Simulate contiguous memory allocation techniques (Worst Fit, Best Fit, First Fit) using a C program.
6. Implement page replacement algorithms (FIFO, LRU, Optimal) in a simulation.
7. Simulate different file organization techniques (Single Level Directory, Two Level Directory).

List of Practicals:

1. Simulate the following CPU scheduling algorithms.

   (a) FCFS (b) SJF (c) Round Robin (d) Priority.

2. Write a C program to simulate producer-consumer problem using Semaphores

3: Write a C program to simulate the concept of Dining-philosophers problem.

4. Simulate MVT and MFT.

5. Write a C program to simulate the following contiguous memory allocation Techniques

   (a) Worst fit (b) Best fit (c) First fit.

6. Simulate all page replacement algorithms

   (a) FIFO (b) LRU (c) OPTIMAL

7. Simulate all File Organization Techniques

8. Simulate all file allocation strategies

   (a) Sequential (b) Indexed (c) Linked.

11. Write a C program to simulate disk scheduling algorithms.

Recommended Books:

1. Operating System Concepts, Silberschatz and Galvin, Wiley India Limited
2. UNIX Concepts and Applications, Sumitabha Das, McGraw-Hill Education
3. Operating Systems, Internals and Design Principles, Stallings, Pearson Education, India
4. Operating System Concepts, Ekta Walia, Khanna Publishing House
5. Modern Operating Systems, Andrew S. Tanenbaum, Prentice Hall of India
6. Operating systems, Deitel & Deitel, Pearson Education, India

*****

MICROPROCESSORS & PERIPHERAL DEVICES

Course Code:	474004
Course Title	Microprocessors and Peripheral Devices
No. of Credits	5 (TH:4,T:0,P:2)

COURSE OUTCOMES: After completion of this course, student will be able to:

1. Understand the evolution and impact of microprocessors on modern society.
2. Gain knowledge of the architecture and functional blocks of microprocessors, particularly the 8085.
3. Develop an understanding of instruction timing, cycles, and programming concepts for the 8085 microprocessor.
4. Learn about memory and I/O interfacing techniques in microprocessors.
5. Understand the concepts and implementation of interrupts in microprocessor systems.
6. Explore different data transfer techniques and peripheral devices used in microprocessor systems.
7. Gain an introduction to advanced microprocessors.

COURSE CONTENTS

1. Microprocessor

Microprocessor, its evolution, function and its impact on modern society.

2. Architecture of a Microprocessor (With reference to 8085 microprocessor)

Concept of Bus, bus organization of 8085, Functional block diagram of 8085 and function of each block, Pin details of 8085 and related signals, De-multiplexing of address/data bus generation of read/write control signals.

3. Instruction Timing and Cycles

Instruction cycle, machine cycle and T-states, Fetch and execute cycle, Timing cycle diagram.

4. Programming (with respect to 8085 microprocessor)

Machine and assembly languages, Machines and Mnemonic codes, Instruction format & Addressing mode, 8085 Instruction set, Instruction set classification, Programming exercises in assembly language.

5. Memories and I/O interfacing

Concept of memory mapping, Address decoding, concept of I/O mapped I/O and memory mapped I/O.

6. Interrupts

Concept of interrupt, Maskable and non-maskable, Edge triggered and level triggered interrupts, Software interrupt, Restart interrupts and its use, Various hardware interrupts of 8085.

7. Data Transfer Techniques

Concept of programmed I/O operations, sync data transfer, async data transfer (hand shaking), Interrupt driven data transfer, DMA, Serial output data, Serial input data

8. Peripheral devices

Intel 8255 PPI, 8253 PIT and 8257 DMA controller

9. Introduction to Advance Microprocessors

Introduction to Pentium series processors and core 2 duo, dual core (core i3, i5, i7)

PRACTICAL OUTCOMES: Upon completion of the course, student will be able to:

1. Gain familiarity with the 8085 microprocessor kit, its keys, and memory map.
2. Understand the steps involved in entering, modifying, and executing programs on the 8085 kit.
3. Develop assembly language programs for performing arithmetic operations on 8-bit numbers.
4. Learn to write assembly language programs for sorting a set of numbers in ascending or descending order.
5. Gain hands-on experience in interfacing peripheral devices such as the 8255, 8253, and 8279 with the 8085.

List of Practicals:

1. Familiarization of different keys of 8085 microprocessor kit and its memory map
2. Steps to enter, modify data/program and to execute a program on 8085 kit
3. Writing and execution of ALP for addition and subtraction of two 8 bit numbers
4. Writing and execution of ALP for multiplication and division of two 8 bit numbers
5. Writing and execution of ALP for arranging 10 numbers in ascending/descending order
6. Interfacing exercise on 8255 like LED display control
7. Interfacing exercise on 8253 programmable interval timer
8. Interfacing exercise on 8279 programmable KB/display interface like to display the hex code of key pressed on display

Recommended Books:

1. Microprocessor Architecture, Programming & Applications with 8080/8085 by Ramesh S Gaonkar, Willey Eastern Ltd. New Delhi
2. Introduction to Microprocessor by Mathur, Tata McGraw Hill Education Pvt. Ltd. New Delhi
3. Microprocessor and Microcontrollers by Dr B P Singh, Galgotia Publications, New Delhi
4. Microprocessor and Applications by Badri Ram: Tata McGraw Hill Education Pvt Ltd , New Delhi
5. Microprocessor and Microcomputers by Refiquzzaman, Prentice Hall of India Ltd., New Delhi
6. Digital Logic and Computer Design by Mano, M Morris; Prentice Hall of India, New Delhi
7. Digital Electronics and Applications by Malvino Leach; Publishers McGraw Hill, New Delhi
8. Digital Integrated Electronics by Herbert Taub and Donals Sachilling; Prentice Hall of India Ltd., New Delhi
9. Digital Electronics by Rajaraman; Prentice Hall of India Ltd., New Delhi
10. Digital Electronics and Microprocessor by Rajiv Sapra, Ishan Publication, Ambala

******

OBJECT ORIENTED PROGRAMMING USING JAVA

Course Code:	474005
Course Title	Object Oriented Programming Using Java
No. of Credits	5 (TH:4,T:0,P:2)

COURSE OUTCOMES: After completion of this course, student will be able to:

1. Understand the fundamentals of object-oriented programming and distinguish it from procedure-oriented programming.
2. Implement various language constructs such as variables, types, data types, operators, iteration statements, and conditionals.
3. Create and work with classes, objects, constructors, and packages in Java.
4. Understand and implement inheritance, access control, abstract classes, and interfaces.
5. Apply polymorphism through method and constructor overloading and overriding.
6. Handle exceptions effectively and utilize multithreading concepts for concurrent programming.

COURSE CONTENTS

1. Introduction and Features:

Fundamentals of object oriented programming – procedure oriented programming Vs. object oriented programming (OOP), Object oriented programming concepts – Classes, object, object reference, abstraction, encapsulation, inheritance, polymorphism, Introduction of eclipse (IDE) for developing programs in Java

2. Language Constructs:

variables, types and type declarations, data types : Integer, floating point type, character, Boolean, all Operators, iteration and jump statement, if then else clause; conditional expressions, input using scanner class and output statement, loops, switch case, arrays, methods.

3. Classes and Objects:

Class fundamentals, constructors, declaring objects (Object & Object Reference), creating and accessing variables and methods, static and non-static variables/methods defining packages, Creating and accessing a package, Importing packages, Understanding CLASSPATH, auto boxing, String , String Buffer

4. Inheritance:

Definition of inheritance, protected data, private data, public data, constructor chaining, order of invocation, types of inheritance, single inheritance, multilevel inheritance, hierarchical inheritance, hybrid inheritance, access control (Private Vs Public Vs Protected Vs Default)

5. Abstract Class and Interface:

Defining an interface, difference between classes and interface, Key points of Abstract class & interface, difference between an abstract class & interface, implementation of multiple inheritance through interface.

6. Polymorphism:

Method and constructor overloading, method overriding, up-casting and down-casting.

7. Exception Handling:

Definition of exception handling, implementation of keywords like try, catches, finally, throw& throws, built in exceptions, creating own exception sub classes importance of exception handling in practical implementation of live projects

8. Multithreading:

Difference between multi-threading and multi-tasking, thread life cycle, creating threads, thread priorities, synchronizing threads.

PRACTICAL OUTCOMES: Upon completion of the course, student will be able to:

1. Understand the use of the super and this keywords and their significance in class implementation.
2. Gain proficiency in designing classes with inheritance and static methods for modeling real-world scenarios.
3. Demonstrate knowledge of abstract methods and classes and their application in program design.
4. Develop a comprehensive understanding of string manipulation and the implementation of string-related operations.
5. Apply class hierarchies and inheritance concepts to model and manipulate objects in specific domains, such as cars or employees.
6. Learn to effectively handle exceptions and implement error-handling mechanisms in programs.

List of Practical:

1. WAP to create a simple class to find out the area and perimeter of rectangle and box using super and this keyword.
2. WAP to design a class account using the inheritance and static that show all function of bank (withdrawal, deposit).
3. WAP to design a class using abstract methods and classes.
4. WAP to design a string class that perform string method (equal, reverse the string, change case).
5. Consider we have a Class of Cars under which Santro Xing, Alto and Wagon R represents individual Objects. In this context each Car Object will have its own, Model, Year of Manufacture, Colour, Top Speed, etc. which form Properties of the Car class and the associated actions i.e.,

object functions like Create(), Sold(), display() form the Methods of Car Class.

6. In a software company Software Engineers, Sr. Software Engineers, Module Lead, Technical Lead, Project Lead, Project Manager, Program Manager, Directors all are the employees of the company but their work, perks, roles, responsibilities differs. Create the Employee base class would provide the common behaviors of all types of employee and also some behaviors properties that all employee must have for that company.
7. Using the concept of multiple inheritance create classes: Shape, Circle, Square, Cube, Sphere, Cylinder. Your classes may only have the class variable specified in the table below and the methods Area and/or Volume to output their area and/or volume.

Class	Class Variable	Constructor	Base class
Shape	String name	Shape ()	
Circle	double radius	Circle (double r, String n)	Shape
Square	double side	Square (double s, String n)	Shape
Cylinder	double height	Cylinder (double h, double r, String n)	Circle
Sphere	None	Sphere (double r, String n)	Circle
Cube	None	Cube (double s, String n)	Square

8. WAP to handle the exception using try and multiple catch block.
9. WAP that implement the Nested try statements.
10. WAP to create a package that access the member of external class as well as same package.
11. WAP that show the partial implementation of interface.
12. WAP to create a thread that implement the Runnable interface.

Text & Reference Books:

1. Programming with Java: A Primer; E. Balagurusamy
2. Head First Java, O-REILLY, Kathy Sierra & Bert Bates.
3. OCA Java SE Programmer I Certification Guide , Wiley Publisher , Mala Gupta
4. PROGRAMMER'S GUIDE TO JAVA SE 8 , Pearson , Khalid E Mughal
5. e-books/e-tools/relevant software to be used as recommended by AICTE/ UBTER/ NITTTR.

*****

‘Elective 1-1’
ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Course Code:	454007
Course Title	Artificial Intelligence & Machine Learning
No. of Credits	4 (TH:4,T:0,P:0)

COURSE OUTCOMES: At the end of this course, the student will be able to:

1. Understand the history and foundations of artificial intelligence, including its origins and key milestones in its development.
2. Gain knowledge and proficiency in problem-solving techniques in AI.
3. Develop an understanding of adversarial search in decision support systems and technologies.
4. Acquire knowledge of representation, reasoning, expert systems, and the basics of planning in AI.
5. Learn the basics tools and techniques used in machine learning.

COURSE CONTENTS

Unit - I : Introduction

History & foundations of AI, Problem solving: Uninformed and informed Search.

Unit - II : Adversarial Search

Two players games, games with uncertainty; Decision support systems and technologies; Knowledge representation, Reasoning.

Unit - III : Machine Learning Basics

Decision trees, Ensemble learning, Reinforcement learning, Evolutionary computation, Neural networks, Visualization.

Unit - IV :

Basic idea of Linear regression, concept of SSE; gradient descent; closed form; normal equations; features.

Unit - V :

Classification problems; Decision boundaries; Probability and classification, Bayes optimal decisions.

References:

1. Russell, Norvig, Artificial intelligence: A modern approach, 2nd edition. Pearson/Prentice Hall.

2. M.C. Trivedi, A classical approach to Artificial Intelligence, Khanna Publishing House, New Delhi (2018)

3. V.K. Jain, Machine Learning, Khanna Publishing House, New Delhi (2018)

4. Ethem Alpaydin, Introduction to Machine Learning, Second Edition,

5. http://mitpress.mit.edu/catalog/item/default.asp?ttype=2&tid=12012.

*****

‘Elective 1-2’
SOFT COMPUTING

Course Code:	454008
Course Title	Soft Computing
No. of Credits	4 (TH:4,T:0,P:0)

COURSE OUTCOMES : At the end of the course, the student will be able to:

1. Classify and differentiate problem solving methods and tools.
2. Apply A*, AO*, Branch and Bound search techniques for problem solving.
3. Formulate an optimization problem to solve using evolutionary computing methods.
4. Design and implement GA, PSO and ACO algorithms for optimization problems in Mechanical Engineering.
5. Apply soft computing techniques for design, control and optimization of Manufacturing systems.

COURSE CONTENTS

Unit - I : Introduction

Soft Computing, Difference between Hard and Soft computing, Requirement of Soft computing, Major Areas of Soft Computing, Applications of Soft Computing.

Unit - II : Neural Networks

Introduction to Neural Network, Learning rules and various activation functions, Single layer Perceptrons, Back Propagation networks, Architecture of Backpropagation (BP) Networks, Neural Network.

Unit - III : Fuzzy Systems

Fuzzy Control Systems, Fuzzy Classification.

Unit - IV : Genetic Algorithm

History of Genetic Algorithms (GA), Working Principle, Various Encoding methods.

Unit - V : Hybrid Systems

Sequential Hybrid Systems, Auxiliary Hybrid Systems, Embedded Hybrid Systems.

Text & Reference Books:

1. Tettamanzi Andrea, Tomassini and Marco, Soft Computing Integrating Evolutionary, Neural and Fuzzy Systems, Springer, 2001.
2. Elaine Rich, Artificial Intelligence, McGraw Hill, 2/e, 1990.
3. Kalyanmoy Deb, Multi-objective Optimization using Evolutionary Algorithms, John Wiley and Sons, 2001.

*****

‘Audit Course’
ESSENCE OF INDIAN TRADITIONAL KNOWLEDGE

Course Code	AS401
Course Title	Essence of Indian Traditional Knowledge
No. of Credits	0 (TH:2,T:0,P:0)

COURSE OUTCOMES: After completion of this course, student will be able to:

1. Develop a comprehensive understanding of the essence of Indian knowledge and tradition.
2. Explore the rich philosophical systems of ancient India and their relevance today.
3. Gain familiarity with the Vedic literature and scriptures, and appreciate their wisdom.
4. Analyze Indian epics and mythology to understand their cultural and spiritual significance.
5. Learn and apply principles of yoga, meditation, and mindfulness for personal well-being.
6. Discover the principles and practices of Ayurveda and natural healing for holistic health.

COURSE CONTENTS

1. Introduction to Indian Knowledge and Tradition
2. Ancient Indian Philosophical Systems
3. Vedic Literature and Scriptures
4. Indian Epics and Mythology
5. Yoga, Meditation, and Mindfulness Practices
6. Ayurveda and Natural Healing Systems
7. Indian Classical Arts and Music
8. Indian Architecture and Sculpture
9. Indian Festivals and Rituals
10. Ethical and Moral Values in Indian Culture

References /Suggested Learning Resources:

1. "Indian Philosophy: A Very Short Introduction" by Sue Hamilton
2. "The Vedas: An Introduction to Hinduism's Sacred Texts" by Roshen Dalal
3. "The Ramayana: A Shortened Modern Prose Version of the Indian Epic" by R.K. Narayan
4. "The Upanishads" translated by Eknath Easwaran
5. "Autobiography of a Yogi" by Paramahansa Yogananda
6. "Ayurveda: The Science of Self-Healing" by Dr. Vasant Lad.

*****

MINOR PROJECT WORK

Course Code:	AS402
Course Title	Minor Project Work
No. of Credits	2 (TH:0,T:0,P:4)

OBJECTIVE:

The Minor Project work is an integral part of the Engineering Diploma program, designed to provide students with an opportunity to apply theoretical knowledge gained throughout their studies to real-world engineering challenges. This module aims to foster creativity, problem-solving abilities, and practical skills essential for successful engineering professionals.

PRACTICAL OUTCOMES: After undergoing the minor project work, the student will be able to:

1. Understand the practical applications of engineering concepts in real-world scenarios.
2. Develop hands-on experience in designing, implementing, and testing engineering projects.
3. Enhance problem-solving and critical thinking skills through project execution.
4. Improve documentation and presentation skills for effective project communication.

GENERAL GUIDELINES:

1. Introduction to Minor Projects

- Overview of the module's purpose and objectives
- Importance of practical application in engineering
- Understanding the project life cycle and its stages

2. Project Ideation and Proposal Development

- Identifying engineering problems and project ideas
- Formulating clear project objectives and scope
- Developing a comprehensive project proposal

3. Project Planning and Management

- Creating a project plan with defined milestones and timelines
- Resource allocation and budgeting for the project
- Risk assessment and mitigation strategies

4. Engineering Design and Analysis

- Principles of engineering design and problem-solving
- Conducting feasibility studies and simulations (if applicable)
- Engineering analysis techniques and tools

5. Prototyping and Implementation

- Hands-on development of project prototypes
- Conducting experiments and data collection
- Troubleshooting and problem-solving during implementation

6. Project Documentation and Reporting

- Techniques for effective project documentation
- Writing comprehensive project reports and design documentation
- Organizing and presenting project data

7. Project Presentation and Communication

- Principles of effective communication in engineering
- Preparing engaging & informative project presentations
- Addressing questions & feedback during the presentation

8. Project Evaluation and Assessment

- Criteria for evaluating project success and achievement of objectives
- Conducting fair and unbiased project assessments
- Peer evaluations and constructive feedback.

ACTIVITIES AND EXECUTION GUIDELINES

1. Project Proposal Submission:

Students will submit their project proposals to the assigned mentors. The proposals should be well-structured, indicating the project's significance, expected outcomes, resources required, and a preliminary plan of action.

2. Project Execution:

During this period, students will work on their projects under the guidance of their mentors. They are encouraged to employ innovative techniques and apply engineering principles to achieve project objectives successfully.

3. Project Documentation:

Students will submit their final project reports and related documentation. The documentation should encompass all project phases, methodologies, experimental data, analysis, and outcomes.

4. Project Presentation:

Each student will deliver a comprehensive presentation to a panel of evaluators, showcasing their project's key aspects, results, and conclusions.

ASSESSMENT CRITERION

1. Project Proposal and Objective (10%)

Students are required to submit a comprehensive project proposal outlining the problem statement, objectives, scope, and methodology of the project. This component will account for 10% of the total marks.

2. Project Implementation (60%)

The core of the assessment will be based on the successful implementation of the project. Students will be evaluated on their ability to execute the project plan, adhere to timelines, and demonstrate practical engineering skills. This segment will carry 60% of the total marks.

3. Documentation (15%)

Proper documentation is vital to effective project management and communication. Students will be evaluated on the clarity, completeness, and organization of their project reports, design diagrams, code (if applicable), and any other relevant material. This component will contribute 15% of the total marks.

4. Project Presentation (15%)

Communication and presentation skills are crucial for engineers to articulate their ideas effectively. Students will be assessed based on their ability to present their project's objectives, methodology, results, and conclusions in a clear and concise manner. This segment will be worth 15% of the total marks.

The Minor Project module is a pivotal component of the Engineering Diploma program that provides students with hands-on experience, encourages critical thinking, and prepares them for real-world engineering challenges. By adhering to the module guidelines and distribution of marks, students can excel in their projects and demonstrate their engineering prowess effectively.

*****